

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-259479

(43)Date of publication of application : 22.09.2000

(51)Int.Cl. G06F 12/00
G06F 13/00

(21)Application number : 11-060851

(71)Applicant : NEC CORP

(22)Date of filing : 08.03.1999

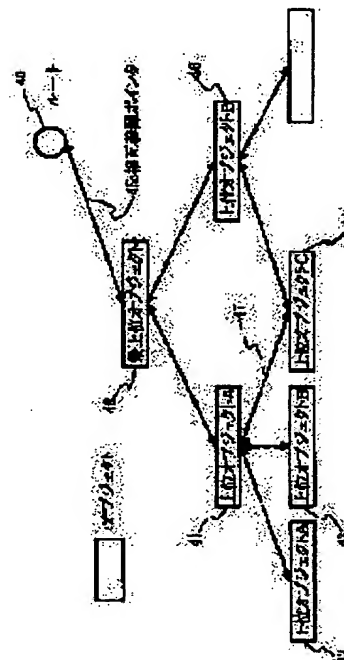
(72)Inventor : TSUNEDA MASAYA

(54) METHOD FOR REALIZING DIRECTORY SERVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To realize normalized attribute information management having no contradiction between a plurality of objects all the time in providing a directory service.

SOLUTION: This service realizing method associates data corresponding to an upper object in a directory service managed and stored by a database management system with data corresponding to a lower object by using a mutual reference pointer 45 and provides a means to easily specify an upper object 41 from a lower object 42 by tracking the pointer 45 from the lower object 42 to the upper object 41 in a reverse direction. Consequently, attribute information that is common between upper and lower objects is managed and stored in a single place of the object 41 in order to obtain attribute information that does not have contradiction between objects having common attribute information all the time and consequently attribute information can be returned as if to be attribute information of the object 42 itself by tracking the pointer 45 from the object 42 in a reverse direction.



LEGAL STATUS

[Date of request for examination] 01.03.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's]

decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2000-259479
(P2000-259479A)

(43)公開日 平成12年9月22日(2000.9.22)

(51)Int.Cl. ⁷	識別記号	F I	ターマート*(参考)
G 0 6 F 12/00	5 4 5	G 0 6 F 12/00	5 4 5 A 5 B 0 8 2
13/00	3 5 7	13/00	3 5 7 Z 5 B 0 8 9

審査請求 有 請求項の数4 O L (全 8 頁)

(21)出願番号 特願平11-60851

(22)出願日 平成11年3月8日(1999.3.8)

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 恒田 正哉

東京都港区芝五丁目7番1号 日本電気株式会社内

(74)代理人 100108578

弁理士 高橋 昭男 (外3名)

Fターム(参考) 5B082 BA11 EA01 EA07 EA11 GA11

GA15 HA03 HA09

5B089 JA12 KA11 KA13 KB00 KC19

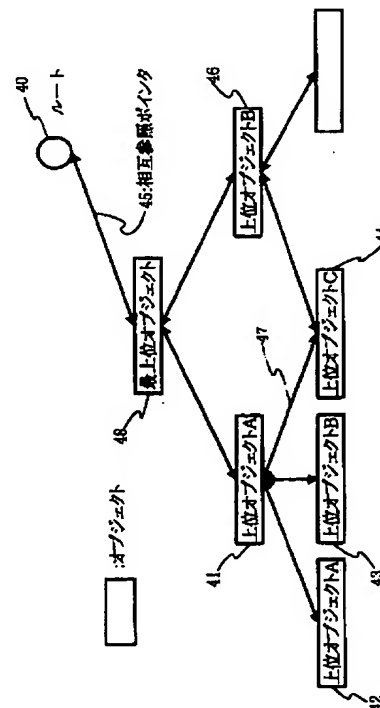
KC52

(54)【発明の名称】 ディレクトリサービスの実現方法

(57)【要約】

【課題】ディレクトリサービスを提供する上で常に複数のオブジェクト間で矛盾の無い正規化された属性情報の管理を実現する。

【解決手段】データベース管理システムに管理・格納されたディレクトリサービスにおける上位オブジェクトに該当するデータと、下位オブジェクトに該当するデータとの間を相互参照ポインタ45を用いて関連付け、相互参照ポインタ45を下位オブジェクト42から上位オブジェクト41へ逆方向に辿ることによって、容易に下位オブジェクト42から上位オブジェクト41を特定する手段を提供する。この結果、常に共通する属性情報を持つオブジェクト間で矛盾の無い属性情報を得るために、上下間で共通する属性情報については上位オブジェクト41の一ヶ所で管理・格納することとし、下位オブジェクト42からは相互参照ポインタ45を逆方向に辿ることによって、あたかも下位オブジェクト42自身の属性情報であるかのように返すことを可能とする。



【特許請求の範囲】

【請求項 1】 データベース管理システムを用いたディレクトリサービスの実現方法において、データベース管理システム内に設けた相互参照ポインタを用いてオブジェクト間の上下関係を表すことによって、ツリー構造情報をデータベース管理システム内で直に管理・格納することを特徴とするディレクトリサービスの実現方法。

【請求項 2】 相互参照ポインタを逆方向に辿ることによって、上位オブジェクトの属性情報をあたかも自分自身の属性情報として利用することを特徴とする請求項 1 記載のディレクトリサービスの実現方法。

【請求項 3】 請求項 2 記載の属性情報の返し方を用いることによって、常に関連する複数のオブジェクト間で、矛盾の無い正規化された属性情報を得ることを特徴とする請求項 2 記載のディレクトリサービスの実現方法。

【請求項 4】 データベース管理システムを用いたディレクトリサービスの実現方法において、ユーザを表すオブジェクトにアクセス制限のための情報を格納し、データベース管理システム内に設けられたオブジェクト間の上下関係を表す相互参照ポインタを逆方向に辿り、上位のオブジェクトが保有するアクセス制御情報と比較することによって、各ユーザの更新可能範囲をサブツリー単位で制限することを特徴とするディレクトリサービスの実現方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はデータベース管理システムを用いたディレクトリサービスの実現方法に係り、特にディレクトリサービスシステムにおける属性の管理方式に特徴を有するディレクトリサービスの実現方法に関する。

【0002】

【従来の技術】 従来のディレクトリサービスシステムにおける属性管理方式では、すべてのオブジェクトが個々に独立して属性情報を管理している。例えば特開平7-129454号公報「オブジェクト指向データベースを用いたディレクトリシステムの構築方法」にその一例が記載されている。この公報に記載された属性情報の管理方式によって管理されるオブジェクトの構成を図 6 に示し、この図を用いて従来の方式について説明する。

【0003】 図 6 は、属性型 a, b, c の 3 個の属性を保持するオブジェクト 10 に対する属性情報の構成例を示している。この例では属性型 c はオプション属性である。従来の属性情報管理方式では、ディレクトリサービスにおける一属性をオブジェクト指向でのインスタンス変数 100, 101, 102 として捉え、これらインスタンス変数の集合をオブジェクト 10 が保持する属性の集合として管理して

いる。

【0004】 図 6 において、インスタンス変数 100, 101, 102 はそれぞれ属性型 a, b, c に対応するものであり、符号 103, 104, 105 は属性型 a, b, c の属性値への各アクセスの手続きを示すものである。また、オブジェクト 11, 12, 13 は、それぞれインスタンス変数 100, 101, 102 に設定されたポインタによって示されるものであり、オブジェクト 11 が属性値 A110 と属性値アクセスの手続き 111 を保持し、オブジェクト 12 が属性値 I120 および属性値 U121 と属性値アクセスの手続き 122 を保持し、そして、オブジェクト 13 が属性値アクセスの手続き 131 を保持している。

【0005】 また、図 7 に上記公報に記載されたオブジェクト間の構造情報の格納形式を示し、この図を用いて説明する。ディレクトリサービスにおけるオブジェクト間の構造情報は、ツリー構造であることが規定されている。そこで従来の方式では、ルートと呼ばれる特別なオブジェクト 20 を起点とし、図 6 に示される個々のオブジェクト情報を一方向のポインタ 25 で関連付けることによってオブジェクト間の上下関係を表現する。この結果、データベース管理システム内部においても直接ツリー構造を形作ることが可能となる。例えば、上位オブジェクト 21 の直接下位オブジェクトとして、下位オブジェクト A, B, C (22, 23, 24) のそれぞれが上位オブジェクト 21 から一方向ポインタで連結されることによって、オブジェクト間の上下関係が表され、ツリー構造が規定される。しかし、他のオブジェクトの情報を参照する手段、特に下位オブジェクトから上位オブジェクトへの参照手段は考慮されていない。

【0006】 次に、図 8 に構造情報の格納形式の具体例を示し、この図を用いて説明する。図 8 において部署オブジェクト 31 に所属するユーザとして、ユーザオブジェクト A, B, C (32, 33, 34) が一方向ポインタで連結され、格納されている。ここで、例えば「所属部署名」といった属性を管理する場合、部署オブジェクト 31 のみならずユーザオブジェクト A, B, C (32, 33, 34) のすべてのオブジェクトにおいて全く同じ属性値を管理する必要が生じる。また、部署オブジェクト 31 の部署名が変更になった場合、この部署オブジェクト 31 の下位に属するすべてのユーザオブジェクト（ユーザオブジェクト A, B, C (32, 33, 34)）の所属部署名属性に対応する属性値データを更新を行わなければならない。このとき、すべてのユーザオブジェクトの所属部署名属性に対応する属性値データを更新しなければ、その内容に食い違いが発生し、関連するオブジェクト間で矛盾が生じることになる。さらに、ユーザオブジェクト C (34) の所属部署が、部署オブジェクト A (31) から部署オブジェクト B (36) に異動した場合、一方向ポインタ 37 を切り、新たに一方向ポインタ 38 を用いて連結した上で、部署オブジェクト C (34) の所属部署名属性に対応する属性値データを書き換

えなければならない。

【0007】以上のように、従来の技術においては、たとえ複数のオブジェクトに共通する属性情報であっても、個々のオブジェクトが独立して属性情報を管理する機能しか保持していないために、その属性情報が更新された場合には関連するすべてのオブジェクトを更新しなければならない。その結果、このような更新作業は非常に手間がかかり、そのための時間も多く必要としていた。さらに、複数のオブジェクトが個々に属性情報を管理するため、データベース管理システムが提供するディレクトリサービスの情報を記録する媒体の容量が増す原因ともなっていた。

【0008】

【発明が解決しようとする課題】以上のように従来の技術における第一の問題点は、複数のオブジェクトに共通する属性値であっても、関連するすべてのオブジェクトのインスタンス変数で管理しなければならなかった点である。この理由は、すべてのオブジェクトが独立して個々の属性情報を管理していたためである。

【0009】第二の問題点は、複数のオブジェクトに共通する属性値に対する更新要求に対して、関連するすべてのオブジェクトのインスタンス変数の内容を更新する必要があったという点である。この理由は、第一の問題点の理由と同じである。

【0010】第三の問題点は、複数のオブジェクトに共通する属性値に対する参照要求に対して、常に情報の食い違いの無い、正規化された情報を得られる保証が無かった点である。この理由は、第一の問題点の理由と同じである。

【0011】第四の問題点は、他のオブジェクトの属性情報を有効に利用する手段が無かったことである。この理由は、従来の属性管理方式では他のオブジェクトの属性情報の利用手段について、何ら考慮されていなかったためである。

【0012】本発明の目的は、ディレクトリサービスを提供する上で常に複数のオブジェクト間で矛盾の無い正規化された属性情報の管理を実現することである。

【0013】また、本発明の他の目的は、データベース管理システムが実現するディレクトリサービスの情報を記録する媒体の記憶容量を削減することである。

【0014】

【課題を解決するための手段】上記の課題を解決するため、請求項1記載の発明は、データベース管理システムを用いたディレクトリサービスの実現方法において、データベース管理システム内に設けた相互参照ポインタを用いてオブジェクト間の上下関係を表すことによって、ツリー構造情報をデータベース管理システム内で直に管理・格納することを特徴としている。また、請求項2記載の発明は、相互参照ポインタを逆方向に辿ることによって、上位オブジェクトの属性情報をあたかも自分自身

の属性情報として利用することを特徴としている。また、請求項3記載の発明は、請求項2記載の属性情報の返し方を用いることによって、常に関連する複数のオブジェクト間で、矛盾の無い正規化された属性情報を得ることを特徴としている。

【0015】また、請求項4記載の発明は、データベース管理システムを用いたディレクトリサービスの実現方法において、ユーザを表すオブジェクトにアクセス制限のための情報を格納し、データベース管理システム内に設けられたオブジェクト間の上下関係を表す相互参照ポインタを逆方向に辿り、上位のオブジェクトが保有するアクセス制御情報と比較することによって、各ユーザの更新可能範囲をサブツリー単位で制限することを特徴としている。

【0016】以上のように、本発明は、データベース管理システム（以下、DBMS）に管理・格納されたディレクトリサービスにおける上位オブジェクトに該当するデータと、下位オブジェクトに該当するデータとの間を相互参照ポインタを用いて関連付け、DBMS内にディレクトリサービスにおけるツリー構造そのものを直に実現し、相互参照ポインタを下位から上位へ逆方向に辿ることによって、容易に下位オブジェクトから上位オブジェクトを特定する手段を提供するものである。この構成によれば、例えば、複数のオブジェクト間に共通する情報を、関連するオブジェクト中のトップノードである一つのオブジェクトで管理・格納することができ、また、下位オブジェクトに対してその属性へのアクセス要求が伝えられたときには、相互参照ポインタを逆方向に辿ることによって、上位オブジェクトが管理・格納する属性情報にアクセスし、あたかも下位オブジェクトが保持する属性であるかのように振る舞うことができる。

【0017】

【発明の実施の形態】〔構成の説明〕本発明の実施形態について図面を参照して説明する。図3は本発明の一実施形態に係わるディレクトリサービスのモデルを示すブロック構成図である。

【0018】図3に示すようにディレクトリサービスを実現する構成としては、ユーザが希望するディレクトリサービスにおける各種要求を発行し、その操作結果を受け取るクライアント機能60と、要求された各種要求を実現し、操作結果を返すサーバ機能61から構成される。これら二つの機能は同一の計算機上で構成されても良いし、通信回線によって結ばれた別々の計算機上で構成されても良い。

【0019】クライアント60から発行された要求はサーバ内のサーバコア部610で解析され、要求を満たすために必要な操作を、実際に情報を管理・格納するDBMS（612）上で実現する情報管理部611に対して伝えられる。情報管理部611はDBMS（612）上でディレクトリ操作を実現するためのインターフェイスを有し、DBMS（612）内で

5

管理・格納された情報を操作し、その操作結果をサーバコア部610に返す。情報管理部611からの操作結果を受け取ったサーバコア部610は、ディレクトリサービスにおける操作結果をクライアント60に返す。

【0020】〔動作の説明〕次に、本実施形態の動作について図1を参照して説明する。本発明では、ルートと呼ばれる特別なオブジェクト40を起点とし、上位オブジェクトと下位オブジェクトとの間を相互参照ポインタ45を用いて連結することによって、ディレクトリサービスの構造情報をDBMS(612)内に構築する。ここで相互参照ポインタ45は、DBMSに管理・格納されたディレクトリサービスにおける上位オブジェクトに該当するデータと、下位オブジェクトに該当するデータとの間を関連付けるものであって、上位オブジェクトと下位オブジェクトの間で相互に各オブジェクトの保持する属性情報を参照可能とするものである。この結果、上位オブジェクトA(41)から下位オブジェクトA,B,C(42,43,44)を辿ることを可能とすると共に、下位オブジェクトA,B,C(42,43,44)から上位オブジェクトA(41)へ逆方向に辿ることも容易に可能となる。さらには、下位から上位へと逆方向に相互参照ポインタを順に辿り続けることによって、最上位オブジェクト48がもつ属性情報にアクセスすることも容易に可能となる。

【0021】また上下間で共通の属性情報を保持する場合、上位オブジェクトA(41)でのみ、その属性情報を管理・格納する。そして下位オブジェクトA(42)に対してその属性情報の参照要求が伝えられたときには、相互参照ポインタを逆方向に辿ることによって、上位オブジェクトA(41)が管理・格納する属性情報に対しアクセスすることによって、あたかも下位オブジェクトA(42)が保持する属性であるかのように振る舞う。

【0022】図4にその際の動作の流れを示し、図1, 図3, 図4を参照して説明する。なお、本実施の形態において、参照要求の対象となるオブジェクトとしては、図6に示すものと基本的な構成が同様なものを用いることとする。ただし、本実施形態では、各オブジェクトに、相互参照ポインタに関する手続きが格納されている。

【0023】まず、クライアント60からの参照要求に対して、手続き70に従いサーバコア部610において参照すべきオブジェクトとなる対象オブジェクトを特定し(ステップ700)、ステップ701において情報管理部611に対し、対象オブジェクトの参照すべき属性の参照要求を発行する。そして参照要求を受け取った情報管理部611では、手続き71に従い対象オブジェクトをDBMS612内部で特定し(ステップ710)、そのオブジェクトに対して対象属性の参照要求を発行する(ステップ711)。さらに、参照要求を受け取ったオブジェクト(図6のオブジェクト10に対応)内に規定される属性値アクセス手段

(手続き103,104,105に対応)に設定されている手続き7

6

2に従いステップ720において上位オブジェクトA(41)への相互参照ポインタを逆に辿ることにより、対象属性を参照し、その参照結果を情報管理部611に返す。参照結果を受け取った情報管理部611はさらにサーバコア部610に返し、サーバコア部610はクライアント60に参照結果を返す。

【0024】また、図1において、下位オブジェクトC(44)の所属が、上位オブジェクトA(41)から上位オブジェクトB(46)に異動した場合には、相互参照ポインタ47の上位オブジェクトへの参照先を上位オブジェクトB(46)に向けるだけで、それ以降の下位オブジェクトC(44)に対する属性参照要求において、新しい上位オブジェクトB(46)の属性情報を利用することが可能となる。

【0025】〔具体的構成の説明〕図2に本発明における構造情報の格納形式の具体例を示し、図2および図4を参照しながら具体的な構成および動作を説明する。なお、図2において符号50~57で示す各構成は、図1で符号40~47で示した各構成に対応している。

【0026】図2において部署オブジェクト51に所属するユーザとして、ユーザオブジェクトA,B,C(52,53,54)が相互参照ポインタで連結され、管理・格納されている。ここで、ユーザオブジェクトA(52)の所属部署名属性を参照するケースを考える。このとき本発明では、部署オブジェクト51のみが部署名を表す属性情報を管理・格納しているが、この部署オブジェクト51の下位に属するユーザオブジェクトA,B,C(52,53,54)は部署名を表す属性情報を管理・格納していなくても、ユーザオブジェクトA,B,C(52,53,54)の所属部署名属性を参照することが可能となる。すなわち、図4の手続き71におけるステップ710において、ユーザオブジェクトA(52)に対する参照要求であることが特定され、ステップ711において、ユーザオブジェクトA(52)に対する所属部署名属性のアクセスが発行される。これを受けたユーザオブジェクトA(52)は、所属部署名を参照するために、相互参照ポインタを逆に辿り上位オブジェクトである部署オブジェクトA(51)が保持する部署名情報をアクセスし、あたかもユーザオブジェクトA(52)自身がその属性を保持するかのように振る舞うことによって、所属部署属性を参照することが可能となる。

【0027】また、部署オブジェクトA(51)の部署名が変更になった場合、この部署オブジェクトA(51)の所属部署名属性に対応する属性値を更新するだけで、それ以降のユーザオブジェクトの所属部署名属性の参照要求に対しても自動的に新しい属性値が反映される。この結果、部署オブジェクトA(51)の下位に属するユーザオブジェクトA,B,C(52,53,54)は、常に同じ所属部署名を参照することが可能となり、関連する複数のオブジェクト間で属性値の矛盾が生じることはない。

【0028】また、ユーザオブジェクトC(54)の所属

部署が、部署オブジェクトA (51) から部署オブジェクトB (56) に異動した場合、相互参照ポインタ57の上位オブジェクトへの参照先を部署オブジェクトB (56) に向けるだけで、それ以降のユーザオブジェクトC (54) の所属部署名属性の参照要求に対して、新しい所属部署名として部署オブジェクトB (56) が返される。

【0029】〔発明の他の実施の形態〕本発明の応用例として、ユーザの更新可能範囲を制限するアクセス制御機能を実現する例について、図5にこのアクセス制御機能の具体例を示し、その動作を説明する。

【0030】この実施の形態においては、ユーザオブジェクト内にアクセス権限を表す値を格納し、この値を基準として相互参照ポインタを逆方向に辿る規則を設定した上で、辿った先の上位オブジェクトをトップノードとするサブツリーを対象ユーザの更新可能範囲として認識する一方で、それ以外のオブジェクトに対する更新要求を禁止する。

【0031】図5において、ユーザオブジェクト80は、アクセス権限を表す値（アクセス権コード）としてインスタンス変数800一つに相互参照ポインタを逆方向に辿る回数を保持しているものとする。このとき、ディレクトリサービスを利用するユーザに該当するユーザオブジェクト80を特定し、そのアクセス権限を表す値が“2”の場合、相互参照ポインタ85を逆方向に2回辿り、部署オブジェクト81を介して辿った先の上位オブジェクトである上位部署オブジェクト82をトップノードとするサブツリーを更新可能範囲86として認識する。このとき、更新可能範囲外のオブジェクトである最上位部署オブジェクト83に対して更新要求を発行した場合、エラーが返される。

【0032】同様にアクセス権限を表す値が“3”の場合、相互参照ポインタ85を逆方向に3回辿り、辿った先の上位オブジェクトである最上位部署オブジェクト83をトップノードとするサブツリーを更新可能範囲として認識する。

【0033】このようにして、本実施の形態では、ユーザの更新可能範囲を制限するアクセス制御機能を実現することができる。

【0034】

【発明の効果】第一の効果は、複数のオブジェクトに共通する属性値を一個所で管理することが可能となることである。この結果、複数のオブジェクトに共通する属性値は常に正規化された同一の情報を得ることが可能となる。この理由は、複数のオブジェクトに共通する属性値の参照要求において、それらに共通した上位オブジェクトへの逆ポインタを辿ることによって属性情報を取得するためである。

【0035】第二の効果は、更新要求を満たすために必要な作業ステップが削減されることである。この理由は、複数のオブジェクトに共通する属性値の更新要求に

において、トップノードのオブジェクトの属性値を更新した後、それ以降の関連するすべてのオブジェクトに対する参照要求では、それらに共通した上位オブジェクトへの逆ポインタを辿ることによって自動的に新しい属性値が反映されるためである。

【0036】第三の効果は、ディレクトリサービスの情報を記録する媒体の記憶容量が削減される。この理由は、複数のオブジェクトに共通する属性値を一個所で管理するために、これに対応した情報を記録する上でも一度記録するだけで良いためである。

【図面の簡単な説明】

【図1】本発明におけるツリー構造情報格納モデルを示す図。

【図2】図1のより具体的なツリー構造情報格納モデルの一例を示す図。

【図3】本発明の一実施形態に係わるディレクトリサービスモデルを示すブロック図。

【図4】本発明における下位オブジェクトから上位オブジェクトへの属性参照手順を示す図。

【図5】本発明におけるツリー構造情報格納モデルを応用したアクセス制御方式を示す図。

【図6】ディレクトリサービスの構築手段となる属性情報格納モデルを示す図。

【図7】ディレクトリサービスの構築手段となる従来のツリー構造情報格納モデルを示す図。

【図8】図7のより具体的なツリー構造情報格納モデルの一例を示す図。

【符号の説明】

10～13：オブジェクト

20：ルート

21：上位オブジェクト、22～24：下位オブジェクト

25：一方向ポインタ

30：ルート

31, 36：部署オブジェクト

32～34：ユーザオブジェクト

35, 37, 38：一方向ポインタ

40：ルート

41, 46：上位オブジェクト

42～44：下位オブジェクト

45, 47：相互参照ポインタ

50：ルート

51, 56：部署オブジェクト

52～54：ユーザオブジェクト

55, 57：相互参照ポインタ

60：クライアント

61：サーバ

70～72：手続き

80：ユーザオブジェクト

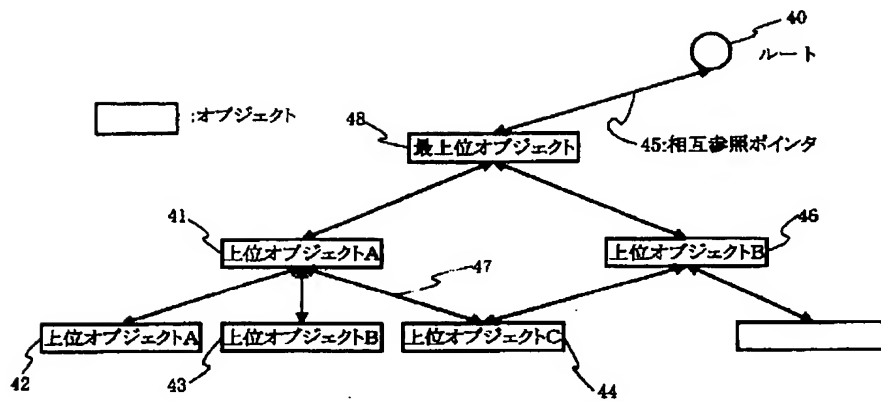
81～83：部署オブジェクト

50' 84：ルート

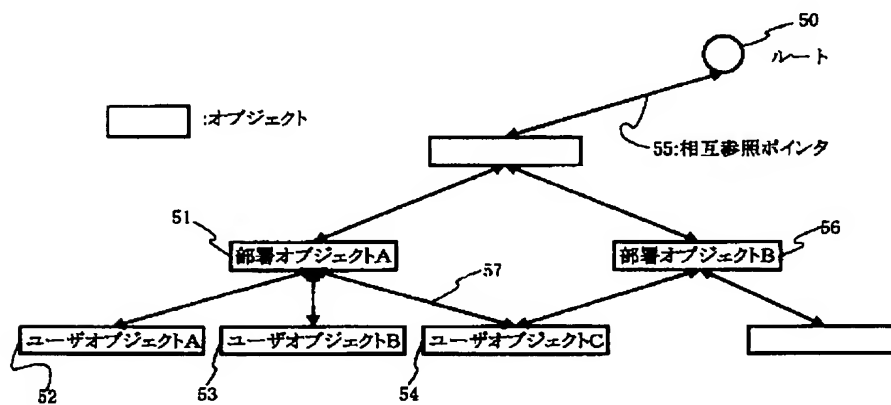
85: 相互参照ポインタ

86: 更新可能範囲

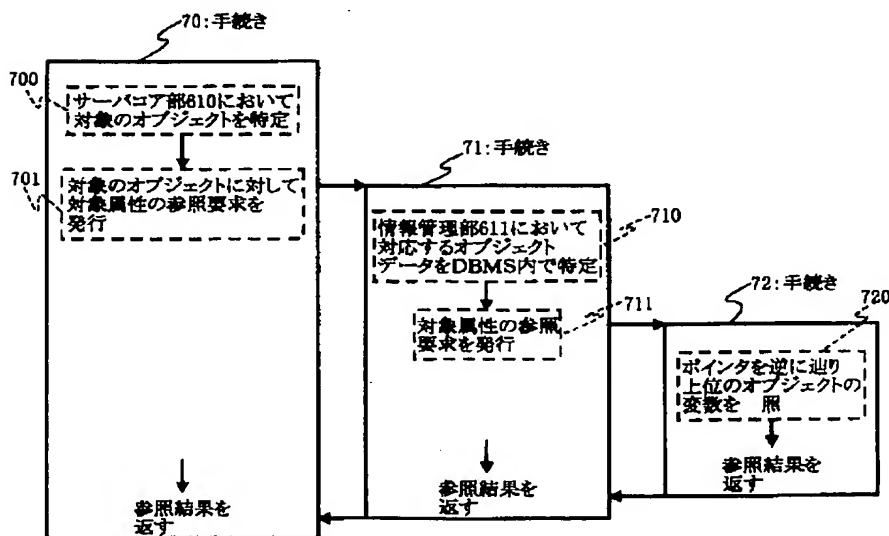
【図1】



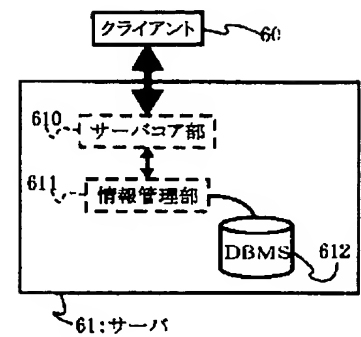
【図2】



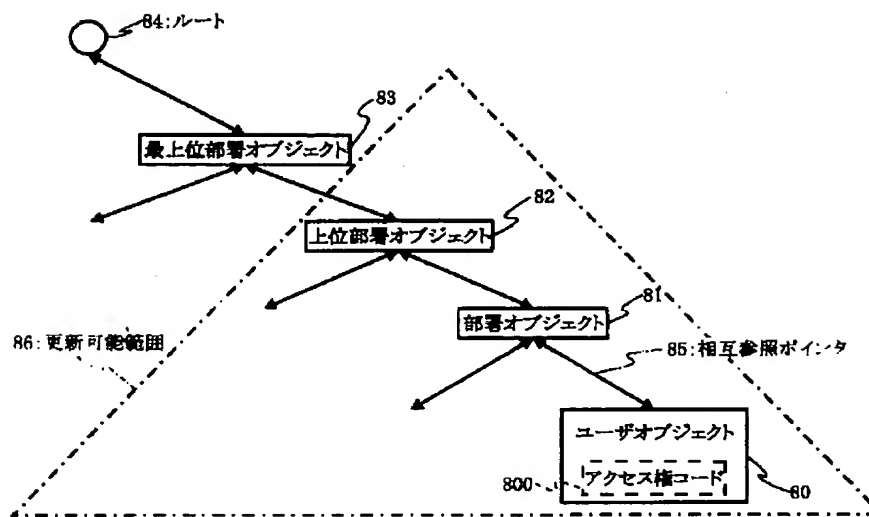
【図4】



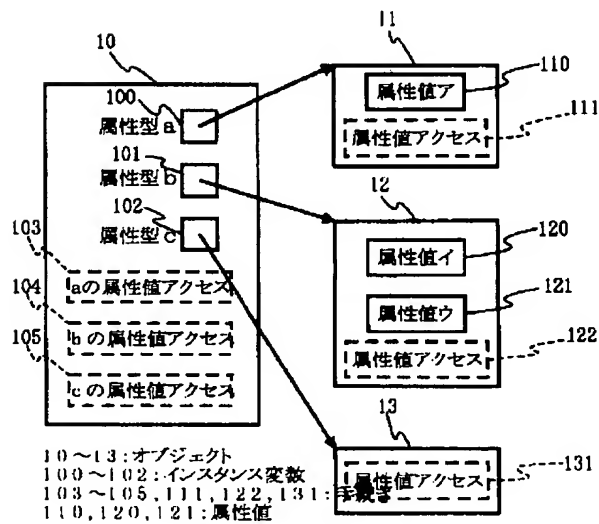
【図3】



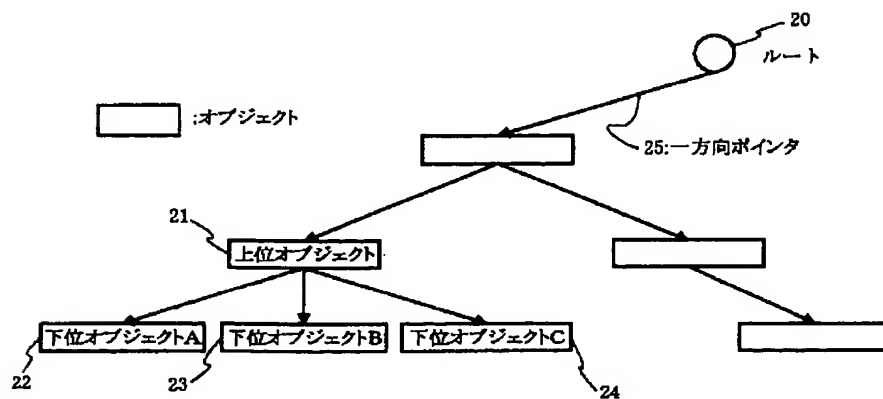
【図5】



【図6】



【図7】



【図 8】

